MovingFashion: a Benchmark for the Video-to-Shop Challenge –Supplementary Material–

Marco Godi*[1], Christian Joppi*[1], Geri Skenderi*[1], and Marco Cristani[1,2]

[1]Department of Computer Science, University of Verona
[2]Humatics Srl, Verona, Italy
{marco.godi,christian.joppi,geri.skenderi}@univr.it
marco.cristani@{univr,humatics}.it

1. Supplementary Material Outline

The supplementary material is organized as follows: in the present file, the following topics are covered:

- **MovingFashion additional details (Sec. 2)**: Additional details on the collection and creation of the MovingFashion dataset;
- **SEAM Match-RCNN computational complexity (Sec. 3)**: Computational complexity of SEAM Match-RCNN;
- **Additional experiments (Sec. 4)**: Additional results, certifying the superiority of SEAM Match-RCNN w.r.t. state-of-the-art single image street-to-shop approaches and their natural extensions, dealing with multiple street images as input. This is complementary to Table 2 and Table 6 of the main paper, where SEAM Match-RCNN is compared against the native multi-image approaches NVAN [4], VKD [5], MGH [8], Asymnet [1] and extensions of the Match-RCNN [2];
- **Future perspectives (Sec. 5)**: Future perspectives of our work, motivating further research on the video-to-shop challenge and on our new dataset MovingFashion in particular.

The `videos.zip` file[1] contains videos (mp4 codec) where the analysis of attention scores computed on some MovingFashion sequences is reported. This gives an interpretation of high and low attention values, corroborating what was written in Sec. 5.3 of the main paper: high attention comes when clothing items are captured without (auto) occlusions on full body shots or when they are zoomed, showing their entire shape, possibly portraying discriminative details (sharp logos for example).

2. MovingFashion Additional Details

2.1. Image and video collection

In this section we give further details on the process of data collection and annotation of Moving Fashion.

Regarding the data collected from the Net-A-Porter website, the data labeling was a long, yet linear process, the only issue being the removal of classes not in the DeepFashion2 taxonomy, in particular *shoes* (deserving of a specific fashion taxonomy) and *jewelry* (due to the lack of a shared and widely accepted aesthetical taxonomy). For the remaining classes, the association to the specific DF2 taxonomy was direct.

Plenty more work was required for the data downloaded from Instagram. In order to to download the data, the Instaloader[2] tool was employed. We manually selected a list of hashtags and profiles with a lot of content, i.e. a lot of videos paired with fashion products for sale. Through the use of the tool, we downloaded posts containing videos only based on the previously mentioned hashtags and profiles. The layout of these videos was standard for the vast majority of them: the frame was divided vertically in two parts, one with just a still picture of the shop product and one with the video itself.

We manually annotated these videos by following these steps:

- We checked that the product actually appears in the video, since in some cases the item never appears or appears very briefly in the frame; sometimes the item is in a different color than the one in the shop image.
- We drew a bounding box around the area of the shop item(s), taking care to include as few other items as possible.

*indicates equal contribution

[1] publicly available here `https://bit.ly/3uKmjhh`

[2] `https://instaloader.github.io/`

Method	**MovingFashion**				**Regular-MovingFashion**				**Hard-MovingFashion**			
	T-1	**T-5**	**T-10**	**T-20**	**T-1**	**T-5**	**T-10**	**T-20**	**T-1**	**T-5**	**T-10**	**T-20**
SFM-First	0.20	0.43	0.52	0.63	0.21	0.44	0.53	0.64	0.16	0.41	0.52	0.62
SFM-1qrt	0.25	0.53	0.66	0.77	0.29	0.58	0.71	0.82	0.15	0.37	0.51	0.63
SFM-Median	0.23	0.48	0.61	0.75	0.26	0.53	0.66	0.79	0.17	0.33	0.47	0.65
SFM-3qrt	0.21	0.47	0.60	0.72	0.24	0.53	0.66	0.77	0.13	0.29	0.42	0.57
SFM-Last	0.11	0.31	0.41	0.53	0.14	0.35	0.46	0.58	0.05	0.18	0.27	0.36
EPHN-First (2020) [7]	0.15	0.34	0.44	0.53	0.16	0.36	0.46	0.55	0.11	0.27	0.37	0.47
EPHN-1qrt	0.24	0.45	0.55	0.65	0.28	0.51	0.62	0.72	0.13	0.24	0.32	0.42
EPHN-Median	0.27	0.49	0.58	0.66	0.32	0.57	0.67	0.74	0.10	0.24	0.32	0.42
EPHN-3qrt	0.24	0.47	0.55	0.65	0.29	0.55	0.64	0.74	0.09	0.21	0.29	0.40
EPHN-Last	0.17	0.33	0.41	0.49	0.20	0.39	0.47	0.56	0.07	0.15	0.19	0.27
KPM-First (2019) [6]	0.19	0.40	0.51	0.61	0.22	0.45	0.56	0.67	0.09	0.26	0.33	0.45
KPM-1qrt	0.27	0.48	0.60	0.71	0.32	0.56	0.69	0.80	0.12	0.24	0.33	0.45
KPM-Median	0.24	0.48	0.59	0.69	0.27	0.55	0.67	0.78	0.12	0.25	0.35	0.43
KPM-3qrt	0.23	0.46	0.56	0.69	0.27	0.53	0.65	0.76	0.10	0.22	0.28	0.39
KPM-Last	0.16	0.35	0.45	0.55	0.20	0.41	0.53	0.65	0.05	0.14	0.19	0.23
SEAM Match-RCNN	**0.49**	**0.80**	**0.89**	**0.94**	**0.55**	**0.86**	**0.94**	**0.97**	**0.30**	**0.62**	**0.76**	**0.87**

Table 1. SEAM Match-RCNN retrieval results on MovingFashion compared with Single-frame approaches. Note: T-K means Top-K Accuracy.

Method	**MovingFashion**				**Regular-MovingFashion**				**Hard-MovingFashion**			
	T-1	**T-5**	**T-10**	**T-20**	**T-1**	**T-5**	**T-10**	**T-20**	**T-1**	**T-5**	**T-10**	**T-20**
Max Confidence	0.29	0.59	0.72	0.83	0.31	0.63	0.76	0.86	0.21	0.46	0.60	0.71
Max Matching	0.26	0.60	0.74	0.85	0.29	0.65	0.79	0.89	0.17	0.44	0.58	0.74
Average Match-RCNN [1]	0.39	0.73	0.84	0.91	0.43	0.79	0.88	0.94	0.24	0.56	0.70	0.81
Average Descriptor	0.37	0.72	0.86	0.93	0.42	0.78	0.90	0.95	0.21	0.57	0.75	0.85
EPHN-MaxConf (2020) [7]	0.22	0.43	0.55	0.65	0.26	0.50	0.61	0.71	0.10	0.22	0.34	0.44
EPHN-MaxMatching	0.35	0.59	0.67	0.74	0.42	0.68	0.76	0.81	0.14	0.32	0.41	0.52
EPHN-AvgMatching	0.31	0.55	0.64	0.73	0.37	0.64	0.73	0.81	0.11	0.28	0.37	0.48
EPHN-AvgDescriptor	0.22	0.43	0.52	0.61	0.26	0.49	0.58	0.68	0.10	0.24	0.33	0.43
KPM-MaxConf (2019) [6]	0.25	0.47	0.57	0.68	0.30	0.54	0.65	0.77	0.11	0.25	0.32	0.43
KPM-MaxMatching	0.30	0.54	0.66	0.75	0.36	0.61	0.73	0.82	0.13	0.32	0.42	0.53
KPM-AvgMatching	0.34	0.58	0.68	0.77	0.40	0.68	0.78	0.86	0.15	0.28	0.38	0.48
KPM-AvgDescriptor	0.34	0.58	0.69	0.77	0.40	0.68	0.78	0.86	0.15	0.28	0.38	0.48
SEAM Match-RCNN	**0.49**	**0.80**	**0.89**	**0.94**	**0.55**	**0.86**	**0.94**	**0.97**	**0.30**	**0.62**	**0.76**	**0.87**

Table 2. SEAM Match-RCNN retrieval results on MovingFashion compared with Multi-frame approaches. Note: T-K means Top-K Accuracy.

- We drew another bounding box around the area of the video.

Using these annotations we crop the street videos and shop images. This results in pairings, where in some cases we have more than one shop item associated with a street video.

Next, we dealt with duplicates of shop products. In some cases the same product is showcased in multiple videos by different users, but fortunately, the shop image used in such videos is the same. We leveraged this fact to perform a duplicate search for all the shop images. Products that were found to be duplicates were merged, creating pairings where for one shop product multiple videos are associated. To perform this search, for each product we searched for duplicates using a pre-existing tool[3] that employs Perceptual Hash. However we found out that in order to have a very high recall, this process also includes a lot of false positives. To perform a more thorough search, we tried an Image Registration technique using the RANSAC algorithm between each shop image and the duplicate candidates found using the tool. We tried to estimate a Similarity Transform,

[3] `https://github.com/umbertogriffo/fast-near-duplicate-image-search`

Method	**MultiDeepFashion2**			
	T-1	**T-5**	**T-10**	**T-20**
Max Confidence	0.19	0.44	0.54	0.66
Max Matching	0.14	0.45	0.61	0.75
Average Match-RCNN [1]	0.22	0.49	0.63	0.74
Average Descriptor	0.20	0.48	0.60	0.71
EPHN-MaxConf (2020) [7]	0.11	0.19	0.24	0.29
EPHN-MaxMatching	0.11	0.21	0.26	0.33
EPHN-AvgMatching	0.16	0.29	0.34	0.41
EPHN-AvgDescriptor	0.12	0.22	0.27	0.33
KPM-MaxConf (2019) [6]	0.09	0.20	0.25	0.30
KPM-MaxMatching	0.08	0.16	0.21	0.28
KPM-AvgMatching	0.10	0.20	0.25	0.32
KPM-AvgDescriptor	0.13	0.25	0.33	0.40
SEAM Match-RCNN	**0.28**	**0.54**	**0.66**	**0.76**

Table 3. SEAM Match-RCNN retrieval results on MultiDeepFashion2 compared with Multi-frame approaches. Note: T-K means Top-K Accuracy.

to account for translations and scaling (as is the case for these images). We then put a threshold on average pixel difference to separate between duplicates and non duplicates. Since no Python libraries that implement RANSAC are available, it was performed using a custom script.

To make sure that MovingFashion respects the privacy of social media users, we have rendered any face in the videos blurred using a publicly available, face blurring tool[4].

2.2. Tracklet generation

As described in the paper, for all data, noisy tracklet annotations are available. In order to create them:

- Our SEAM Match-RCNN is trained on the data *using only video-image pairing annotations*. This results in a model where the Single-frame Matching Head can be effectively used for precisely tracking each item.
- We use the trained model to build a set of tracklets for each video.
- We manually go over each video and select the tracklets that contain the paired shop item, merging them if they are disjointed (this happens when an item is occluded completely or disappears from the frame and two separate tracklets are built).

The resulting tracklets are then saved. While for our approach, no tracklet annotations are used during training, they are used for all the comparative approaches. They are considered as equivalent to ours (the detector and the tracker are the same). It can be argued that they are actually better than ours as they are produced after the last epoch of training, while for our approach we start with a tracker that has not been trained yet. For the Person Re-ID approaches, the annotations are used to crop out part of the image according to the extracted bounding box. For detection based approaches, the bounding boxes are used as ground truth bounding boxes. The testing tracklets are used by all approaches for evaluation. During the SEAM Match-RCNN evaluation, they are used to select the tracklet among the ones produced automatically by the *tracking procedure*.

3. SEAM Match-RCNN computational complexity

In this Section we discuss the computational complexity of our proposed SEAM Match-RCNN. In particular we focus on the difference between the Single-frame Matching Head and the Multi-frame Matching Head.

3.1. Single-frame Matching Head

Let TF be the time taken for computing features by using the f function and TM the time taken for computing matching between two feature vectors using m.

Given a street image and a shop image, the cost of computing a matching between them, assuming that the detection from the street image has already been chosen in some way (for example by comparing it with a ground truth bounding box) is $2 \times TF + TM$ (features computed for both street and shop are compared).

3.2. Multi-frame Matching Head

When extending to Multi-frame matching, the cost of tracking has to be taken into consideration. Obviously the time taken for feature computation increases linearly with the number of frames sampled from the video.

As $\tilde{f}$ and $\tilde{m}$ are structured in the same way as f and m, we can assert that TF and TM also apply to them. Given a street video sequence from which we sample T frames, the cost of building all the possible tracklets (using the *tracking procedure*, Sec 4-1 of the main paper) is related to the number of detections in each frame K (to simplify notation we assume that there are exactly K detections in each frame). First of all, Single-frame Matching Head features are computed, the time cost is $TF \times K \times T$.

As a reminder, the tracking procedure consists of iteratively repeating the choice of *pivot* and *propagation*. The choice of the pivot is performed by choosing the most confident detection, so its cost is negligible as it is already included in the detection. The propagation consists of doing comparisons between the pivot features and all of the detection features in a frame. For a Single-frame the time necessary for the propagation step is $TM \times K$ (a matching for each detection). This procedure is repeated for all frames resulting in $TM \times K \times T$. This results in a single tracklet, that is excluded from the set of detections for

[4] https://github.com/ORB-HD/deface

the following iterations. As the iteration is repeated until there are no more detections, we can assume that repeating the propagation K times results in a final cost of $TM \times K^2 \times T$. For the whole tracking procedure, the total time is $(TF \times K \times T) + (TM \times K^2 \times T)$.

After tracklets are built, we can assume that the correct tracklet is chosen, for example by using the Intersection over Union with the ground truth tracklets (analogous to selecting the correct bounding box in the Match-RCNN). Given a sequence of detection of length T (length of the video sequence), the cost of computing Multi-frame Matching features is again $TF \times T$. Then self-attetion with the Non-Local Block is performed, resulting in a time cost of $T^2 \times TSA$ (TSA is the cost of computing self-attention between a pair of element in the sequence, usually a simple operation like a dot product). The attention score is then computed for each frame, with a cost of $T \times TA$ (TA is the cost of computing the attention score, in our case a simple linear layer). Finally a weighted average pooling is performed and matching is computed between the aggregated descriptor and the shop feature vector ($TF + TM$). The final cost for aggregation is $(TF \times T) + (T^2 \times TSA) + (T \times TA) + (TF + TM)$.

3.3. Discussion

It is expected that the extension from Single-frame to Multi-frame will come with an increased cost, in relation to the number of frames. The tracking procedure is a necessary step for any possible Multi-frame approach, as detections from each frame need to be grouped in some way. The matching component increases quadratically with the maximum number of detections in each frame and linearly with the number of frames sampled from the sequence.

The aggregation has a term that increases quadratically with the number of frames. For both of these, we have to take into consideration that we usually work with 10 samples and there are rarely many different people and clothing items in a video, so even with a quadratic complexity, the total effective time is relatively small. In our experiments, we never go over 2 seconds for the whole procedure, with the majority of the videos taking about 1 second to process.

4. Additional experiments

In Table 1, we show the results of Single-frame baselines built on top of the Match-RCNN (the main building block of our SEAM Match-RCNN). In particular, SFM-1qrt uses the frame at the first quartile of all the available frames of that sequence, SFM-median uses the median frame and so on. SFM stands for Single-frame match and is a short term for Match-RCNN.

The correspondent baselines are shown, adopting the Deep Kronecker-Product Matching (KPM) [6] and the Easy Positive Triplet Mining approach (EPHN) [7]. The rationale of this choice was to focus on Single-frame Re-Identification approaches and compare them to the Match-RCNN. This was done to enlarge the spectrum of possible comparative approaches, which have open-source code available. The idea of considering Re-ID approaches against street-to-shop techniques was also presented in the DPRNet paper [9].

The inferiority of these baselines with respect of the Multi-frame of Table 2 in the main paper, and in particular with SEAM Match-RCNN, is evident and fully understandable.

Notably, in almost all of the MovingFashion partitions (apart the regular one with EPHN), the ·-1qrt baseline gives the higher results, which seems to be in accord with the best practices in social media video editing, that is, that videos have to deliver their main message within approximately 6 seconds [3].

As additional Multi-frame approaches, Table 2 shows Max Confidence, Max Matching and Average Matching scores when considering the KPM [6] and the EPHN [7] as Single-frame method ingredients, in the same way that Match-RCNN was used to calculate Max Confidence, Max Matching and Average Matching from Table 2 of the main paper.

Even in this case, SEAM Match-RCNN gives the best performance, showing an overall superiority of Match-RCNN as a Single-frame tool to aggregate visual clothing information.

The same applies when it comes to MultiDeepFashion2 where we investigate only Multi-frame policies (Max Confidence, Matching, Avg Matching and Descriptor), since Single-frame policies do not have much sense, as the Single-frames are not part of a single sequence. Even in this case, SEAM Match-RCNN is the best alternative (Table 3).

As additional *qualitative* results, on Fig. 1 results of SEAM Match-RCNN for the Hard-MovingFashion dataset are shown. Two types of considerations can be drawn: the first one is the variability of the videos, which here can be appreciated with more examples. Second, the retrieval results on the right display that SEAM Match-RCNN is capable of finding similar images, among a shop gallery that in some cases contains highly similar items (see for example the light gray trousers).

On Fig. 2 results of SEAM Match-RCNN for the Regular-MovingFashion dataset are shown. Here, on street frames which exhibit more regularities, the shop items are vice versa more insidious than the TikTok ones, since they exhibit a lower variability, see for example the black female dresses of row 6. The same rationale holds for the white shirts and the black paints.

Finally, on Fig. 3 retrieval results on MultiDeepFashion2 are shown. Looking at the retrieval results, one can notice that shop items are way less regular/neutral than the ones on the MovingFashion (which anyway represent a more genuine excerpt of an e-commerce website): at the same time, street frames are often zoomed captures of the object of interest, in general offering a retrieval challenge different than the one on MovingFashion. The strong results obtained by SEAM Match-RCNN prove its versatility in working on a broader set of scenarios.

5. Future perspectives

With SEAM Match-RCNN we showed how the contribution of multiple frames can boost the retrieval accuracy by 33% on MultiDeepFashion2 w.r.t Single-frame approaches and by 69% on the MovingFashion dataset. We also obtained new, state-of-the-art results on all of the benchmarks. Still, much progress has to be made in order to present a new product to the market: looking at the results, the probability of finding the correct shop match within the top 20 ranked shop images is 87% on TikTok/Instagram videos. In order to connect all the dots available within the data, one has to exploit all of the details of the clothing items shown in some of the frames, something which we are currently not able to perform (in fact, we are discarding them with low attention), because they cannot be mapped to the general layout of the clothing item. Therefore, we should probably consider 3D atlases and have a common reference there.

This setup can be attractive for many scenarios, for example: 1) a *casual user* can match a video snippet of a nice outfit he/she has captured with a gallery of products (e.g. Zalando, Amazon, etc.); 2) a *fast fashion company* can measure the similarity of clothing items contained in a viral video, or fashion show, with the items of its catalogue, deciding which item to promote the most; 3) Youtube videos can be automatically processed by *video sharing platforms* to build valuable statistics of popular outfits and discover emerging trends.

References

[1] Zhi-Qi Cheng, Xiao Wu, Yang Liu, and Xian-Sheng Hua. Video2shop: Exact matching clothes in videos to online shopping images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4048–4056, 2017.

[2] Yuying Ge, Ruimao Zhang, Xiaogang Wang, Xiaoou Tang, and Ping Luo. Deepfashion2: A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5337–5345, 2019.

[3] Maxwell Golling. Facebook video ads: Best practices for 2019, 2018.

[4] Chih-Ting Liu, Chih-Wei Wu, Yu-Chiang Frank Wang, and Shao-Yi Chien. Spatially and temporally efficient non-local attention network for video-based person re-identification. In *British Machine Vision Conference*, 2019.

[5] Angelo Porrello, Luca Bergamini, and Simone Calderara. Robust re-identification by multiple views knowledge distillation. In *The European Conference on Computer Vision (ECCV)*, 2020.

[6] Yantao Shen, Tong Xiao, Shuai Yi, Dapeng Chen, Xiaogang Wang, and Hongsheng Li. Person re-identification with deep kronecker-product matching and group-shuffling random walk. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[7] Hong Xuan, Abby Stylianou, and Robert Pless. Improved embeddings with easy positive triplet mining. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2474–2482, 2020.

[8] Yichao Yan, Jie Qin, Jiaxin Chen, Li Liu, Fan Zhu, Ying Tai, and Ling Shao. Learning multi-granular hypergraphs for video-based person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[9] Hongrui Zhao, Jin Yu, Yanan Li, Donghui Wang, Jie Liu, Hongxia Yang, and Fei Wu. Dress like an internet celebrity: Fashion retrieval in videos. In *proceedings of the International Joint Conferences on Artificial Intelligence*, pages 1054–1060, 07 2020.

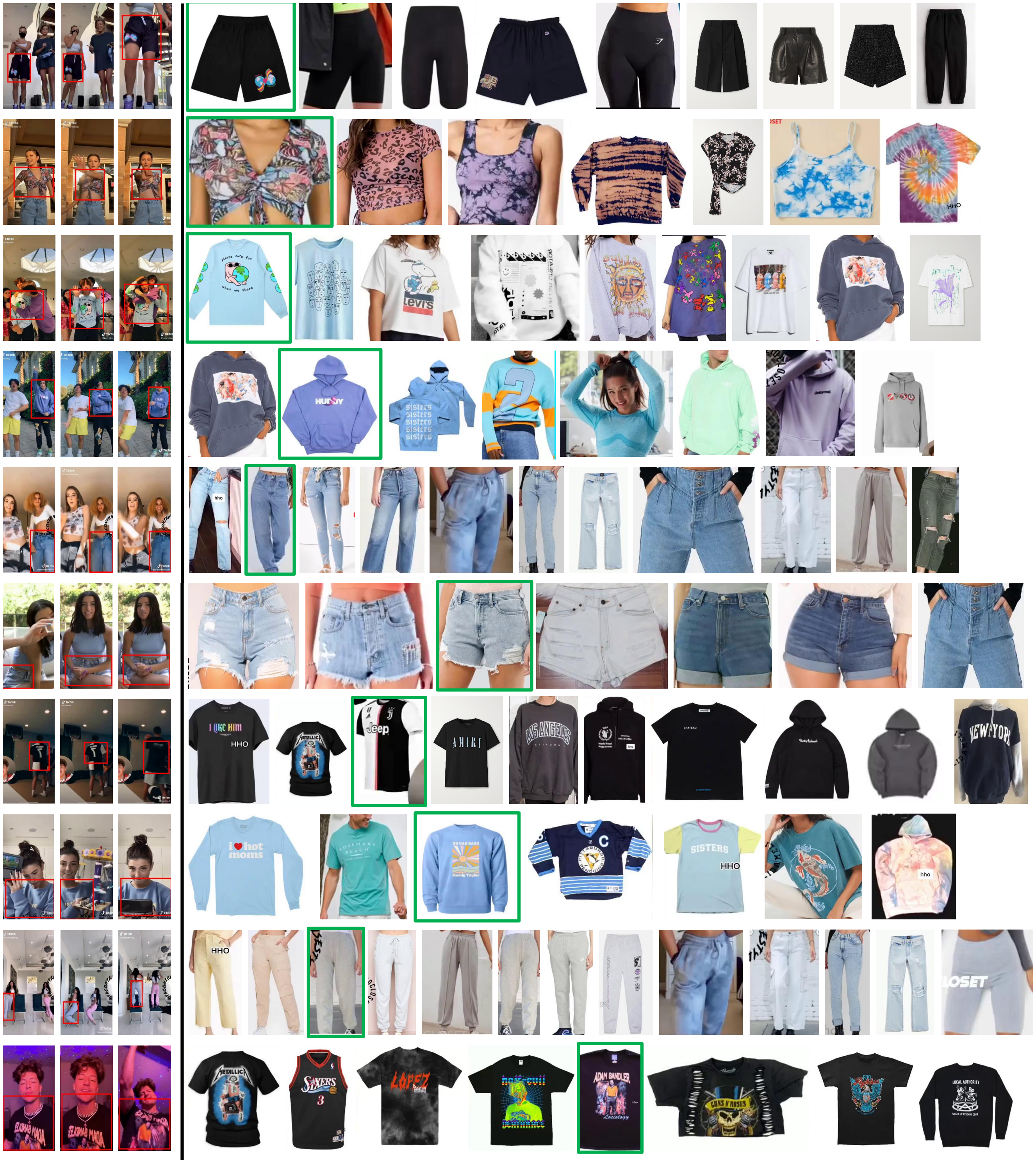

Figure 1. Qualitative retrieval results of SEAM Match-RCNN for the Hard-MovingFashion dataset. On the left, we show 3 frames sampled from the 10 frames used for aggregation. On the right the shop images retrieved starting from the closest match (left). The correct matches are represented with a green border.

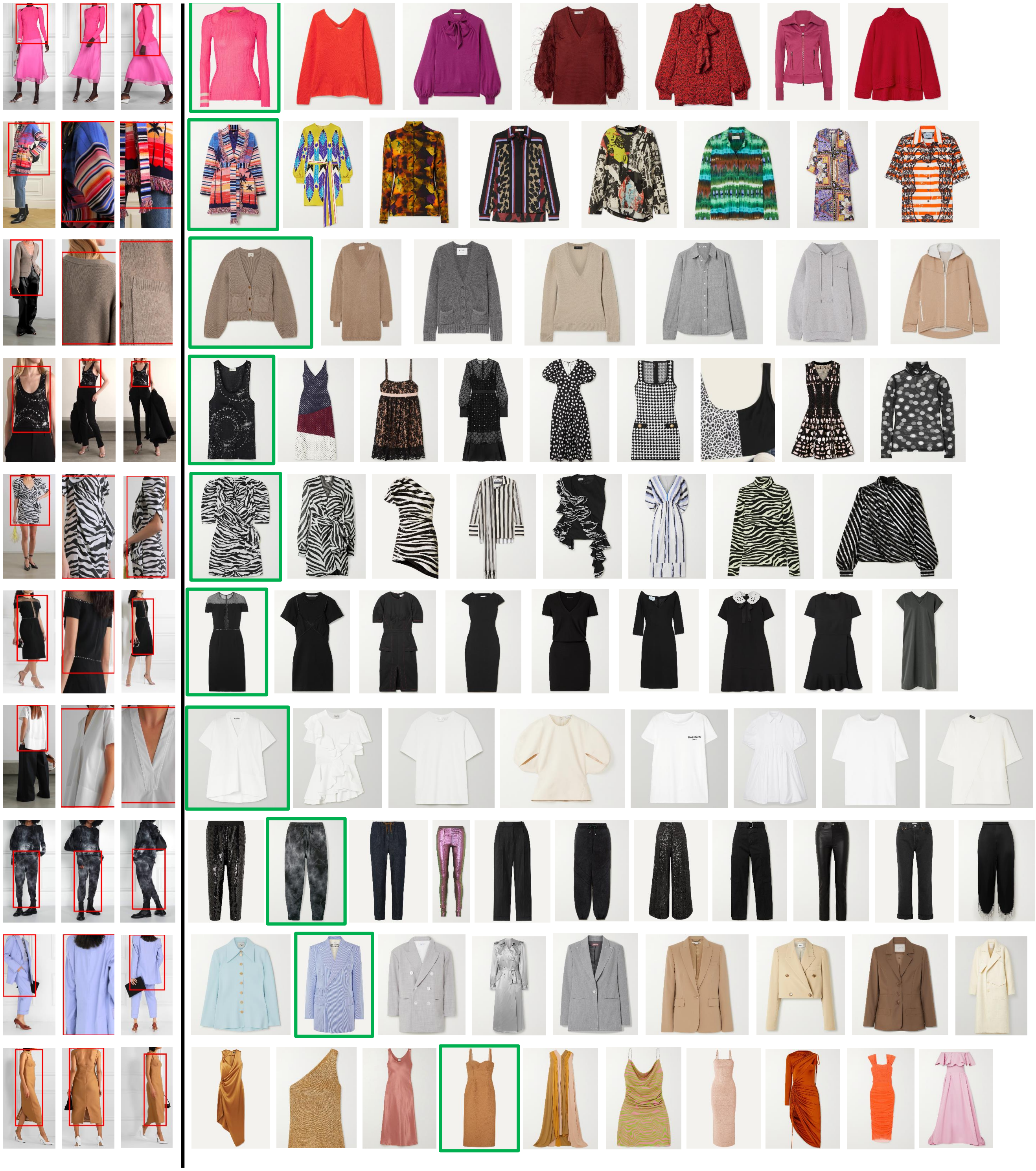

Figure 2. Qualitative retrieval results of SEAM Match-RCNN for the Regular-MovingFashion dataset. On the left, we show 3 frames sampled from the 10 frames used for aggregation. On the right the shop images retrieved starting from the closest match (left). The correct matches are represented with a green border.

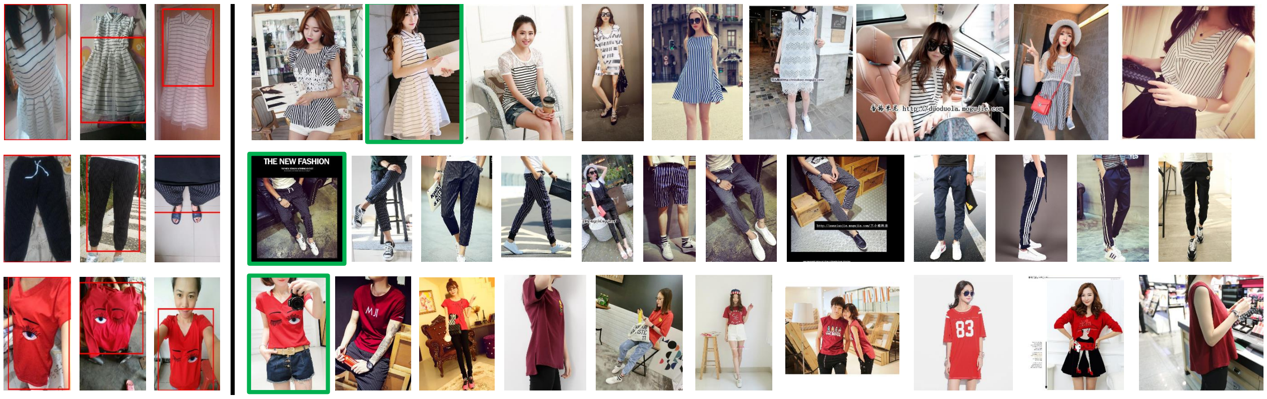

Figure 3. Qualitative retrieval results of SEAM Match-RCNN for the MultiDeepFashion2 dataset. On the left, we show 3 frames sampled from the 10 frames used for aggregation. On the right the shop images retrieved starting from the closest match (left). The correct matches are represented with a green border.